

# BUNDESREPUBLIK DEUTSCHLAND



## Bescheinigung

Die PACT Informationstechnologie GmbH in München/Deutschland hat eine Patentanmeldung unter der Bezeichnung

"IO- und Speicherbussystem für DFPs sowie Bausteinen mit zwei- oder mehrdimensionaler programmierbaren Zellstrukturen"

am 20. Dezember 1996 beim Deutschen Patentamt eingereicht.

Die angehefteten Stücke sind eine richtige und genaue Wiedergabe der ursprünglichen Unterlagen dieser Patentanmeldung.

Die Anmeldung hat im Deutschen Patentamt vorläufig die Symbole G 06 F, G 11 C und H 04 L der Internationalen Patentklassifikation erhalten.

München, den 13. Oktober 1997  
Der Präsident des Deutschen Patentamts  
Im Auftrag

Aktenzeichen: 196 54 595.1

Agurks

Patentanmeldung  
„IO- und Speicherbussystem für DFPs sowie  
Bausteinen mit zwei- oder mehrdimensionaler  
programmierbaren Zellstrukturen“

PACT03 ~→

PACT Informationstechnologie GmbH, München  
Martin Vorbach

18. Dezember 1996

# 1 Hintergrund der Erfindung

## 1.1 Stand der Technik

### 1.1.1 ... in DFP-basierenden Systemen

In DFPs gemäß DE 44 16 881 A1 werden die Leitungen jeder Kantenzelle, das ist eine Zelle, die sich am Rand eines Zellarrays befindet und oftmals direkter Kontakt mit den Anschlüssen des Bausteines haben kann, über die Anschlüsse des Bausteines herausgeführt. Dabei haben die Leitungen keine spezielle Funktion, vielmehr übernehmen sie die Funktion, die in den Kantenzellen beschrieben ist. Werden mehrere DFPs miteinander vernetzt, so werden alle Anschlüsse zu einer Matrix miteinander verbunden.

### 1.1.2 ... in Systemen mit zwei- oder mehrdimensionalen programmierbaren Zellstrukturen (FPGAs, DPGAs)

In Systemen mit zwei- oder mehrdimensionalen programmierbaren Zellstrukturen (FPGAs, DPGAs) wird eine bestimmte Teilmenge von internen Bussystemen und Leitungen der Kantenzellen über die Anschlüsse des Bausteines herausgeführt. Dabei haben die Leitungen keine spezielle Funktion, vielmehr übernehmen sie die Funktion, die in den Kantenzellen beschrieben ist. Werden mehrere FPGAs/DPGAs miteinander vernetzt, so übernehmen die Anschlüsse die Funktion, die hardware- oder softwareseitig implementiert wurde.

## 1.2 Probleme

### 1.2.1 ... in DFP-basierenden Systemen

Der Verdrahtungsaufwand für die Peripherie oder Vernetzung von DFPs ist sehr hoch. Da gleichzeitig der Programmierer dafür zu sorgen hat, daß die entsprechenden Funktionen in die Zellen des/der DFPs integriert werden. Zum Anschluß eines Speichers muß eine Speicherverwaltung in den Baustein integriert werden. Zum Anschluß von Peripherie muß diese unterstützt werden, ebenso wie die Kaskadierung von DFPs entsprechend berücksichtigt werden muß. Der Aufwand ist verhältnismäßig hoch, gleichzeitig geht Fläche auf dem Baustein für die jeweiligen Implementierungen verloren.

### 1.2.2 ... in Systemen mit zwei- oder mehrdimensionalen programmierbaren Zellstrukturen (FPGAs, DPGAs)

Oben beschriebenes gilt ebenfalls für FPGAs und DPGAs, ins besondere, wenn diese zur Implementierung von Algorithmen eingesetzt werden und als numerische (Co)Prozessoren arbeiten.

## 1.3 Verbesserung durch die Erfindung, Aufgabe

Der Verdrahtungsaufwand, ins besondere die Anzahl der Bausteinanschlüsse wird erheblich reduziert. Ein einheitliches Bussystem arbeitet ohne besondere Rücksichtnahme durch den Programmierer. Die Steuerung des Bussystems ist fest implementiert. Ohne besondere Maßnahmen kann an das Bussystem Speicher, sowie Peripherie angeschlossen werden. Ebenfalls können Bausteine mit Hilfe des Bussystems kaskadiert werden.

## 2 Beschreibung der Erfindung

### 2.1 Übersicht über die Erfindung, Abstrakt

Die Erfindung beschreibt ein allgemeines Bussystem, das eine Vielzahl bausteininterner Leitungen zusammenfaßt und als Bündel an die Anschlüsse herausführt. Die Steuerung des Bussystems ist dabei vordefiniert und bedarf keines Einflusses durch den Programmierer. An das Bussystem lassen sich beliebig Speicher, Peripherie oder weitere Bausteine (zur Kaskadierung anschließen). Die Einzelheiten und besondere Ausgestaltungen, sowie Merkmale des erfindungsgemäßen Bussystems sind Gegenstand der Patentansprüche.

### 2.2 Detailbeschreibung der Erfindung

Die nachfolgende Beschreibung umfaßt mehrere Strukturen, die wie bei DFPs, FPGAs, DPGAs, o.ä. üblich von einer Ladelogik aus gesteuert und konfiguriert werden. Dabei können Teile der Ladelogik auf dem Baustein integriert sein. Alternativ besteht die Möglichkeit (Fig. 6/7) die Strukturen durch den Baustein selbst direkt dynamisch umzukonfigurieren oder zu steuern. Dabei können die Strukturen fest auf dem Baustein implementiert sein, oder erst durch Konfiguration und möglicherweise Zusammenschluß mehrere Logikzellen, das sind konfigurierbare Zellen, die einfache logische oder arithmetische Aufgaben gemäß ihrer Konfiguration erfüllen (vgl. DFP, FPGA, DPGA), entstehen.

#### 2.2.1 Bündelung interner Leitungen

Um sinnvolle Busstrukturen zu erhalten, wird eine Mehrzahl interner Leitungen wird zu Bussen (I-BUS<sub>n</sub>; n gibt die Nummer des Busses an) zusammengefaßt. Dabei kann es sich bei den Leitungen um interne Bussysteme oder um Leitungen der Kantenzellen handeln. Die einzelnen Busse werden für Schreibzugriffe auf den externen Bus (E-BUS) über taktsynchrone Latche oder Register (I-GATE-REG) auf Tore geführt, die als Schalter zum E-BUS arbeiten. Eine derartige Einheit wird Ausgabezelle (OUTPUT-CELL) genannt. Der Zugriff auf den E-BUS geschieht derart, daß die einzelnen Latche über die Tore auf den gemeinsamen E-BUS geschaltet werden. Dabei ist immer nur ein Tor geöffnet. Jeder I-BUS<sub>n</sub> besitzt eine eindeutige Kennnummer (n: z.B. I-BUS1, I-BUS976, ...).

Für Lesezugriffe wird der hereinführende E-BUS in taktsynchrone Latche oder Register (E-GATE-REG) zwischengespeichert und dann über die Tore an die I-BUS<sub>n</sub> verteilt. Eine derartige Einheit wird Eingabezelle (INPUT-CELL) genannt. Der Abgriff vom E-BUS geschieht dabei derart, daß ein E-BUS-Transfer in ein oder mehrere E-GATE-REG geschrieben wird. Die E-GATE-REG können dann entweder einzeln oder gemeinsam auf ihre internen Bussysteme freigeschaltet werden.

Dabei können die Schreib-/Lesezugriffe in beliebiger Reihenfolge ablaufen. Es ist unter Umständen sinnvoll die internen Busse I-BUS<sub>n</sub> in zwei Gruppen zu unterteilen, die schreibenden Ausgabebusse IO-BUS<sub>n</sub> und die lesenden Eingabebusse II-BUS<sub>n</sub>.

#### 2.2.2 Adressgenerierung

Für die meisten Zugriffe auf externe Bausteine ist es notwendig Adressen zur Auswahl eines Bausteines oder Teilen eines Bausteines zu generieren. Dabei können die Adressen fest sein, das heißt sie ändern sich nicht (dies ist vor allem bei Peripherieadressen der Fall) oder die Adressen verändern sich bei jedem Zugriff um (meist)

festen Werte (dies ist vor allem bei Speicheradressen der Fall). Zur Generierung der Adressen existieren programmierbare Zähler für die Lesezugriffe und programmierbare Zähler für die Schreibzugriffe. Die Zähler werden durch die Ladelogik, das ist die Einheit, die die, auf einer Zellstruktur basierenden konfigurierbaren Bausteine (DFPs, FPGAs, DPGAs o.ä.) konfiguriert, auf einen Grundwert eingestellt. Bei jedem Zugriff auf das Tor zählt der Zähler je nach Einstellung um einen von der Ladelogik festgelegten Wert nach oben oder unten. Ebenfalls kann jeder Zähler als Register verwendet werden, das bedeutet, bei einem Zugriff wird nicht gezählt, der im Zähler eingestellte Wert ist statisch. Jedem Bustransfer wird als Adresse der Wert des zum Tor gehörenden Zählers zugeordnet. Die Einstellung des Zählers geschieht über ein Einstellungsregister (MODE-PLUREG), das von der Ladelogik beschrieben wird.

### 2.2.3 Masken und Zustände

Jedem Tor ist eine Anzahl Bits im später beschriebenen MODE-PLUREG zugeordnet, die angibt, ob der Tor aktiv ist oder von der Steuerung übergangen wird, also ausmaskiert ist (MASKE). Das bedeutet, der Tor wird beim Durchlaufen aller Tor zum Aufschalten auf das jeweilige Bussystem übergangen.

Folgende Masken-Einträge sind denkbar:

- INPUT-/OUTPUT-CELL immer übergangen
- INPUT-/OUTPUT-CELL nur beim Schreiben übergangen
- INPUT-/OUTPUT-CELL nur beim Lesen übergangen, wenn der E-BUS-MASTER nicht auf die INPUT-/OUTPUT-CELL zugegriffen hat
- INPUT-/OUTPUT-CELL niemals übergangen

Jedem Tor ist ein Zustandsregister, das als RS-Flipflop ausgestaltet sein kann, zugeordnet. Dieses Register zeigt an, ob Daten in das zum Tor gehörenden Register geschrieben wurden.

### 2.2.4 Das MODE-PLUREG

Das MODE-PLUREG kann von der Ladelogik beschrieben und gelesen werden. Es dient zur Einstellung des Bussystems.

Ein möglicher Aufbau des MODE-PLUREG aus Sicht der Ladelogik

Bit l..m	Bit k..l	Bit 2..k	Bit 1	Bit 0
Maske	vordefinierter Wert	Schrittbreite	0=additiv zählen 1=subtraktiv zählen	0=Register 1=Zähler
maskieren	Einstellungen für Adressgenerator			

### 2.2.5 Beschreibung der INPUT-CELL

Es wird unterschieden, ob Daten vom E-BUS in den Baustein gelangen (die dafür notwendige Einheit wird INPUT-CELL genannt) oder ob Daten aus dem Baustein auf den E-BUS gelangen (die dafür notwendige Einheit wird OUTPUT-CELL genannt).

Eine INPUT-CELL kann wie folgt ausgestaltet sein: Ein Latch (I-GATE-REG), das entweder vom externen E-BUS-MASTER oder der bausteineigenen Zustandsmaschine gesteuert wird, dient zur Zwischenspeicherung der Daten, die vom E-BUS erhalten werden. Dabei ist das Taktsignal des Latches an ein (z.B.) RS-Flipflop (SET-REG) geführt, das den Zugriff auf das I-GATE-REG festhält. Hinter dem I-GATE-REG befindet sich ein Tor (I-GATE), das von der Zustandsmaschine gesteuert wird. Über das I-GATE gelangen die Daten aus dem I-GATE-REG auf den I(I)-BUSn.

Des weiteren befindet sich ein programmierbarer auf/abwärts-Zähler in der INPUT-CELL. Er kann nach jedem aktiven Lesezugriff auf den E-BUS durch die Zustandsmaschine gesteuert, um einen einstellbaren Wert hoch- oder herunterzählen. Ebenfalls kann er als einfaches Register dienen. Dieser Zähler generiert die Adressen für Buszugriffe, in denen der Baustein E-BUS-MASTER ist. Die Adressen werden durch ein Tor (ADR-GATE) auf den E-BUS geführt. Das ADR-REG wird von der Zustandsmaschine gesteuert.

Über ein weiteres Tor (STATE-GATE) kann der E-BUS-MASTER den Zustand des SET-REG abfragen. Jede INPUT-CELL besitzt ein MODE-PLUREG, in dem die Ladelogik den Zähler konfiguriert und die INPUT-CELL aus- oder einschaltet (maskiert).

### 2.2.6 Beschreibung der OUTPUT-CELL

Eine OUTPUT-CELL kann wie folgt ausgestaltet sein: Ein Latch (E-GATE-REG), das von der bausteineigenen Zustandsmaschine gesteuert wird, dient zur Zwischenspeicherung der Daten, die vom I-BUS erhalten werden.

Des weiteren befindet sich ein programmierbarer auf/abwärts-Zähler in der OUTPUT-CELL. Dabei ist das Taktsignal des Latches an ein (z.B.) RS-Flipflop (SET-REG) geführt, das den Zugriff auf das E-GATE-REG festhält. Er kann nach jedem aktiven Lesezugriff auf den E-BUS durch die Zustandsmaschine gesteuert, um einen einstellbaren Wert hoch- oder herunterzählen. Ebenfalls kann er als einfaches Register dienen. Dieser Zähler generiert die Adressen für Buszugriffe, in denen der Baustein E-BUS-MASTER ist.

Die Daten des E-GATE-REG, die Adressen und der Zustand des SET-REG werden über ein Tor (E-GATE), das entweder vom externen E-BUS-MASTER oder der bausteineigenen Zustandsmaschine gesteuert wird, auf den geführt. Jede OUTPUT-CELL besitzt ein MODE-PLUREG, in dem die Ladelogik den Zähler konfiguriert und die OUTPUT-CELL aus- oder einschaltet (maskiert).

### 2.2.7 Steuerung des Bussystems

Den einzelnen Toren, Adressgeneratoren und Masken ist eine Steuerung übergeordnet. Diese besteht aus einer einfachen Zustandsmaschine nach dem Stand der Technik. Dabei werden zwei Betriebsarten unterscheiden:

1. Eine Aktive, in der die Zustandsmaschine den internen Bus (I-BUS) und den externen Bus (E-BUS) steuert. Diese Betriebsart wird E-BUS-MASTER genannt, da die Zustandsmaschine die Kontrolle über den E-BUS besitzt.
2. Eine Passive, in der die Zustandsmaschine nur den internen Bus (I-BUS) steuert. Der E-BUS wird von einem anderen externen Baustein kontrolliert. Die Zustandsmaschine reagiert in dieser Betriebsart auf die Anforderungen des externen E-BUS-MASTERS. Diese Betriebsart wird E-BUS-SLAVE genannt.

Die Steuerung verwaltet das E-BUS Protokoll. Dabei ist der Ablauf unterschiedlich, je nachdem, ob die Steuerung als E-BUS-MASTER oder E-BUS-SLAVE arbeitet. Ein Busprotokoll wird in dieser Schrift nicht beschrieben, da eine Vielzahl von Protokollen nach dem Stand der Technik implementiert werden können.

### 2.2.8 E-BUS-MASTER und E-BUS-SLAVE, das EB-REG

Zur Verwaltung des Datenverkehrs auf dem E-BUS existiert das E-BUS-Steuerregister (EB-REG). Es ist in Reihe mit den Toren geschaltet und kann vom E-BUS aus adressiert und angesprochen werden. Über folgende Einträge könnte der Datenaustausch geregelt werden:

- I-WRITE** : zeigt an, daß der I-BUS komplett in die INPUT-/OUTPUT-CELLs geschrieben ist
- I-READ** : zeigt an, daß der I-BUS die INPUT-/OUTPUT-CELLs komplett gelesen hat
- E-WRITE** : zeigt an, daß der E-BUS komplett in die INPUT-/OUTPUT-CELLs geschrieben ist
- E-READ** : zeigt an, daß der E-BUS die INPUT-/OUTPUT-CELLs komplett gelesen hat

Das EB-REG ist immer nur auf der Seite des E-BUS-SLAVE aktiv und der E-BUS-MASTER greift lesend und schreibend darauf zu.

- Alle I-...-Einträge werden vom E-BUS-SLAVE geschrieben und vom E-BUS-MASTER gelesen.
- Alle E-...-Einträge werden vom E-BUS-MASTER geschrieben und vom E-BUS-SLAVE gelesen.

Ein E-BUS-SLAVE kann die Kontrolle über den E-BUS fordern, in dem er das Bit REQ-MASTER in seinem EB-REG setzt. Erkennt der E-BUS-MASTER das Bit REQ-MASTER, so muß er seine Buskontrolle so bald wie möglich abgeben. Dies geschieht, in dem er das Bit MASTER im EB-REG eines E-BUS-SLAVE setzt. Er schaltet daraufhin sofort die E-BUS passiv. Der alte E-BUS-SLAVE wird zum neuen E-BUS-MASTER und der alte E-BUS-MASTER wird zum neuen E-BUS-SLAVE. Der neue E-BUS-MASTER übernimmt die Steuerung des E-BUS. Zur Erkennung des erstens E-BUS-MASTERS nach einer Rücksetzung (RESET) des Systems, existiert ein Anschluß an jedem Baustein, der durch die voreingestellte Polarität angibt, ob der Baustein nach einem RESET E-BUS-MASTER oder E-BUS-SLAVE ist. Der Eintrag MASTER im EB-REG kann auch von der Ladelogik gesetzt und zurückgesetzt werden. Dabei hat die Ladelogik darauf zu achten, daß keinerlei Buskollisionen am EB-BUS entstehen oder laufende Transfers abgebrochen werden.

### 2.2.9 E-BUS-MASTER schreibt Daten in E-BUS-SLAVE

Folgendermaßen kann der E-BUS-MASTER Daten in den E-BUS-SLAVE schreiben:

- Die Datenübertragung beginnt damit, daß die Zustandsmaschine des E-BUS-MASTERS eine OUTPUT-CELL auswählt, die nicht ausmaskiert ist.
- Im I-GATE-REG sind, je nach Ausführung der Zustandsmaschine, bereits Daten gespeichert oder die Daten werden jetzt gespeichert.
- Das Tor wird aktiviert.

- ~ Die gültige Leseadresse wird auf den Bus übertragen.
- ~ Die Daten gelangen auf den E-BUS und werden in das E-GATE-REG des E-BUS-SLAVE gespeichert.
- Dadurch wird das SET-REG im E-BUS-SLAVE aktiviert.
- Das Tor im E-BUS-MASTER wird deaktiviert.
- Der Adresszähler generiert die Adresse für den nächsten Zugriff.
- Der Transfer ist für den E-BUS-MASTER beendet.

Auf der Seite des E-BUS-SLAVE existieren zwei Ausgestaltungsmöglichkeiten die Daten vom Bus in den Baustein zu transferieren:

1. Das Daten-Tor ist immer offen und die Daten gelangen direkt vom E-GATE-REG auf den I-BUSn.
2. Die Zustandsmaschine erkennt, daß SET-REG aktiviert ist und aktiviert das Tor, dadurch kann SET-REG zurückgesetzt werden.

Der E-BUS-MASTER kann dem E-BUS-SLAVE mitteilen, wann ein kompletter Buszyklus beendet ist. (Ein Buszyklus ist als das Übertragen mehrerer Datenworte an unterschiedliche E-GATE-REG definiert, wobei jedes E-GATE-REG genau ein Mal angesprochen werden darf.)

- Dazu setzt der E-BUS-MASTER am Ende eines Buszyklus das Bit E-WRITE im EB-REG des E-BUS-SLAVE.
- Der E-BUS-SLAVE kann darauf reagieren, in dem er die INPUT-CELLs abfragt.
- Hat er alle INPUT-CELLs abgearbeitet, setzt er das Bit I-READ in seinem EB-REG.
- Dabei setzt er E-WRITE und sämtliche SET-REG der INPUT-CELLs zurück.
- Der E-BUS-MASTER kann I-READ abfragen und nach dessen Aktivierung einen neuen Buszyklus beginnen.
- I-READ wird durch Schreiben von E-WRITE oder den ersten Bustransfer zurückgesetzt.

Der E-BUS-SLAVE kann anhand des Zustandes des EB-REG oder der einzelnen SET-REG der INPUT-CELLs auswerten, ob die INPUT-CELLs neu gelesen werden können/müssen.

### 2.2.10 E-BUS-MASTER liest Daten von E-BUS-SLAVE

Aus der Sicht des E-BUS-MASTERs existieren zwei grundlegende Methoden Daten vom E-BUS-SLAVE zu lesen:

1. Verfahren, in dem die E-BUS-Daten direkt auf den I-BUS gelangen:
  - Die Datenübertragung beginnt damit, daß die Zustandsmaschine des E-BUS-MASTERs eine INPUT-CELL auswählt, die nicht ausmaskiert ist.
  - Das I-GATE und das ADR-GATE werden aktiviert.
    - ~ Die gültige Leseadresse wird auf den Bus übertragen.
  - Das I-GATE-REG ist transparent, d.h. es läßt die Daten auf den I-BUSn durch.



- Das Tor im E-BUS-MASTER wird deaktiviert.
  - Der Adresszähler generiert die Adresse für den nächsten Zugriff.
  - Der Transfer ist für den E-BUS-MASTER beendet.
2. Verfahren, in dem die E-BUS-Daten im I-GATE-REG zwischengespeichert werden:
- Die Datenübertragung beginnt damit, daß die Zustandsmaschine des E-BUS-MASTERs eine INPUT-CELL auswählt, die nicht ausmaskiert ist.
  - Das I-GATE und das ADR-GATE werden aktiviert.
  - Die gültige Leseadresse wird auf den Bus übertragen.
  - Das I-GATE-REG speichert die Daten.
  - Das Tor im E-BUS-MASTER wird deaktiviert.
  - Der Adresszähler generiert die Adresse für den nächsten Zugriff.
  - Der E-BUS-Transfer ist für den E-BUS-MASTER beendet.
  - Alle am E-BUS-Transfer beteiligten INPUT-CELLs, dies kann anhand der Masken in den MODE-PLUREG oder dem Zustand der SET-REG festgestellt werden, werden durchlaufen und die Daten auf den jeweiligen I-BUS übertragen.

Für den E-BUS-SLAVE sieht der Zugriff wie folgt aus:

- Durch den E-BUS wird das Tor aktiviert.
- Die Daten und der Zustand des möglicherweise vorhandenen SET-REG gelangen auf den E-BUS.
- Das Tor wird deaktiviert.

Der E-BUS-MASTER kann dem E-BUS-SLAVE mitteilen, wann ein kompletter Buszyklus beendet ist.

- Dazu setzt der E-BUS-MASTER am Ende eines Buszyklus das Bit E-READ im EB-REG des E-BUS-SLAVE.
- Der E-BUS-SLAVE kann darauf reagieren, in dem er die OUTPUT-CELLs neu beschreibt.
- Hat er alle OUTPUT-CELLs abgearbeitet, setzt er das Bit I-WRITE in seinem EB-REG.
- Dabei setzt er E-READ und sämtliche SET-REG der OUTPUT-CELLs zurück.
- Der E-BUS-MASTER kann I-WRITE abfragen und nach dessen Aktivierung einen neuen Buszyklus beginnen.
- I-WRITE wird durch Schreiben von E-READ oder den ersten Bustransfer zurückgesetzt.

Der E-BUS-SLAVE kann anhand des Zustandes des EB-REG oder der einzelnen SET-REG der OUTPUT-CELLs auswerten, ob die OUTPUT-CELLs neu beschrieben werden können/müssen.

### 2.2.11 Anschluß von Speicher, Peripherie und Kaskadierung

Außer der Kaskadierung von gleichen Bausteinen (DFPs, FPGAs, DPGAs) kann an das beschriebene Bussystem Speicher und Peripherie als untergeordnete SLAVE-Baustein (SLAVE) angeschlossen werden. Dabei kann Speicher und Peripherie,

sowie weitere Bausteine (DFPs, FPGAs) gemischt werden. Jeder angeschlossene SLAVE wertet die Adressen auf dem Bus aus und erkennt selbständig ob er angesprochen ist. In diesen Betriebsarten ist der Baustein, der den Speicher oder die Peripherie, also die SLAVE-Bausteine anspricht, der Bus-MASTER (MASTER), d.h. der Baustein steuert den Bus und den Datentransfer. Die Ausnahme bilden intelligente Peripheriebausteine, wie z.B. SCSI-Kontroller, die selbständig Transfers initiieren und durchführen können und daher E-BUS-MASTER sind.

### 2.2.12 Zusammenfassung

Durch das beschriebene Verfahren, lassen sich Bussysteme einfach und leistungsfähig an DFPs und FPGAs anschließen. Dabei können über die Bussysteme sowohl Speicher und Peripherie, als auch weitere Bausteine o.g. Gattung verbunden werden.

Das Bussystem muß nicht ausschließlich in den DFPs, FPGAs und DPGAs implementiert sein. Selbstverständlich ist ein Mischbetrieb dieses Bussystemes mit den herkömmlichen Anschlußstrukturen der Bausteine möglich. Damit können die Vorteile der jeweiligen Technik optimal genutzt werden.

Es sind weitere Ablaufverfahren für das beschriebene Bussystem denkbar. Diese werden jedoch nicht beschrieben, da es sich um freie Ausgestaltungsmöglichkeiten handelt, die vom hier aufzuzeigenden Grundprinzip unabhängig sind.

## 3 Kurzbeschreibung der Diagramme

- Figur 1** : Zeichnung eines Basisbausteines als FPGA Typ A
- Figur 2** : Zeichnung eines Basisbausteines als FPGA Typ B
- Figur 3** : Zeichnung eines Basisbausteines als DFP
- Figur 4** : Leitungsbündelung in FPGAs
- Figur 5** : Leitungsbündelung in DFPs
- Figur 6** : Eine OUTPUT-CELL
- Figur 7** : Eine INPUT-CELL
- Figur 8** : Die Adressgenerierung
- Figur 9** : Komplettes Bussystem mit Steuerung
- Figur 10** : Anschluß von Speicher und Peripherie
- Figur 11** : Das EB-REG
- Figur 12** : Ausführungsbeispiel

## 4 Detailbeschreibung der Diagramme

**Figur 1** zeigt einen FPGA nach dem Stand der Technik. 0101 stellt die internen Bussysteme dar, 0102 beinhaltet eine oder mehrere FPGA-Zelle(n). 0103 sind Teilbusse, die eine Teilmenge von 0101 darstellen und über Schalter (Crossbars) mit 0101 verbunden sind. 0103 kann auch interne Daten von 0102 verwalten, die nicht auf 0101 geschaltet werden. Die FPGA-Zellen sind in einem 2-dimensionalen Array angeordnet. 0104 stellt eine Kantenzone dar, die am Rand des Arrays sitzt und sich somit in nächster Nähe zu den Anschlüssen am Rand des Bausteins befindet.

**Figur 2** zeigt einen weiteren FPGA nach dem Stand der Technik. Diese Ausführungsmöglichkeit arbeitet nicht mit Bussystemen wie 0101 sondern hauptsächlich mit NextNeighbour-Verbindungen (0201), das sind direkte Verbindungen von einer FPGA-Zelle (0203) zu ihrer Nachbarzelle. Es können dennoch globale Bussysteme

(0202) existieren, diese sind jedoch nicht sehr breit. Die FPGA-Zellen oder eine Gruppe von FPGA-Zellen besitzen eine Verbindung zu 0202. Die FPGA-Zellen sind in einem 2-dimensionalen Array angeordnet. 0204 stellt eine Kantenzone dar, die am Rand des Arrays sitzt und sich somit in nächster Nähe zu den Anschlüssen am Rand des Bausteins befindet.

**Figur 3** zeigt einen DFP nach PACT02. Dabei sind die PAE-Zellen (0303) über ein Businterface (0304) auf die Bussysteme (0301) geschaltet. Die Bussysteme 0301 können über einen Busschalter (0302) miteinander verschaltet werden. Die PAE-Zellen sind in einem 2-dimensionalen Array angeordnet. 0305 stellt eine Kantenzone dar, die am Rand des Arrays sitzt und sich somit in nächster Nähe zu den Anschlüssen am Rand des Bausteins befindet.

**Figur 4a** zeigt eine FPGA-Kante nach Figur 1. Außerhalb der Kantenzone (0401) ist eine Mehrzahl von INPUT-/OUTPUT-CELLs (0402) angeordnet, die zu mehreren oder einzeln die internen Bussysteme (0403) mit dem E-BUS (0404) verbinden. Dabei ist die Menge der INPUT-/OUTPUT-CELLs von ihrer eigenen Breite im Verhältnis zur Breite der internen Bussysteme abhängig. 0405 stellt ein EB-REG dar. 0406 stellt eine Zustandsmaschine dar. Von der Zustandsmaschine geht zum EB-REG und jeder einzelnen INPUT-/OUTPUT-CELL ein Bussystem (0407), über das die Zustandsmaschine die INPUT-/OUTPUT-CELLs steuert. Dabei können mehrere 0405 und 0406 existieren, indem je eine Menge an 0402 zu Gruppen zusammengefaßt werden und von je einer 0405 und 0406 verwaltet werden.

**Figur 4b** zeigt eine FPGA-Kante nach Figur 2. Außerhalb der Kantenzone (0411) ist eine Mehrzahl von INPUT-/OUTPUT-CELLs (0412) angeordnet, die zu mehreren oder einzeln die internen Bussysteme (0413) und die direkten Verbindungen der Kantenzone (0417) mit dem E-BUS (0414) verbinden. Dabei ist die Menge der INPUT-/OUTPUT-CELLs von ihrer eigenen Breite im Verhältnis zur Breite der internen Bussysteme (0413) und der Menge der direkten Verbindungen (0418) abhängig. 0415 stellt ein EB-REG dar. 0416 stellt eine Zustandsmaschine dar. Von der Zustandsmaschine geht zum EB-REG und jeder einzelnen INPUT-/OUTPUT-CELL ein Bussystem (0417), über das die Zustandsmaschine die INPUT-/OUTPUT-CELLs steuert. Dabei können mehrere 0415 und 0416 existieren, indem je eine Menge an 0412 zu Gruppen zusammengefaßt werden und von je einer 0415 und 0416 verwaltet werden.

**Figur 5** zeigt eine DFP-Kante nach Figur 3. Außerhalb der Kantenzone (0501) ist eine Mehrzahl von INPUT-/OUTPUT-CELLs (0502) angeordnet, die zu mehreren oder einzeln die internen Bussysteme (0503) mit dem E-BUS (0504) verbinden. Dabei ist die Menge der INPUT-/OUTPUT-CELLs von ihrer eigenen Breite im Verhältnis zur Breite der internen Bussysteme (0503) abhängig. 0505 stellt ein EB-REG dar. 0506 stellt eine Zustandsmaschine dar. Von der Zustandsmaschine geht zum EB-REG und jeder einzelnen INPUT-/OUTPUT-CELL ein Bussystem (0507), über das die Zustandsmaschine die INPUT-/OUTPUT-CELLs steuert. Dabei können mehrere 0505 und 0506 existieren, indem je eine Menge an 0502 zu Gruppen zusammengefaßt werden und von je einer 0505 und 0506 verwaltet werden.

**Figur 6** zeigt eine OUTPUT-CELL 0601. Außerhalb 0601 befindet sich das EB-REG (0602) und die Zustandsmaschine (0603), sowie ein Tor (0604), das die Zustandsmaschine auf den E-BUS (0605) schaltet, sofern sie E-BUS-MASTER ist. Auf das EB-REG kann über den E-BUS (0605), den I-BUS (0613) und den Ladelogikbus (0609) zugegriffen werden. Weiterhin kann bei einem Rücksetzen des Bausteins (Reset) über einen externen Anschluß (0614), der aus dem Baustein herausgeführt ist, das MASTER-Bit gesetzt werden. Die Zustandsmaschine (0603) greift schreibend und lesend auf 0602 zu. In der OUTPUT-CELL befindet sich ein Multiplexer (0606), der die Steuerung des E-GATE (0607) entweder dem E-BUS-MASTER oder der Zustandsmaschine (0603) zuordnet. Das MODE-PLUREG (0608) wird über den Ladelogikbus (0609) oder den I-BUS (0613) eingestellt und konfiguriert den Adresszähler (0610), sowie die Zustandsmaschine (z.B. ausmaskieren der OUTPUT-CELL). Werden Daten des I-BUS (0613) in das I-GATE-REG (0611) gespeichert, wird der Zugriff im SET-REG (0612) vermerkt. Der Zustand von 0612 ist über 0607 am E-BUS abfragbar. Ein lesender Zugriff (das E-GATE 0607 ist aktiviert) setzt 0612 zurück. Die von 0610 generierten Adressen und die Daten von 0611 werden über das Tor 0607 auf den E-BUS übertragen. Es besteht die Möglichkeit die OUTPUT-CELL statt über die Ladelogik über den Baustein (DFP, FPGA, DPGA, o.ä.) selbst dynamisch umzukonfigurieren und zu steuern. Hierzu dient der der I-BUS-Anschluß an das EB-REG (0602) und das MODE-PLUREG (0608).

**Figur 7** zeigt eine INPUT-CELL 0701. Außerhalb 0701 befindet sich das EB-REG (0702) und die Zustandsmaschine (0703), sowie ein Tor (MASTER-GATE) (0704), das die Zustandsmaschine auf den E-BUS (0705) schaltet, sofern sie E-BUS-MASTER ist. Auf das EB-REG kann über den E-BUS (0705), den I-BUS (0713) und den Ladelogikbus (0709) zugegriffen werden. Weiterhin kann bei einem Rücksetzen des Bausteins (Reset) über einen externen Anschluß (0714), der aus dem Baustein herausgeführt ist, das MASTER-Bit gesetzt werden. Die Zustandsmaschine (0703) greift schreibend und lesend auf 0702 zu. In der INPUT-CELL befindet sich ein Multiplexer (0706), der die Steuerung des E-GATE-REG (0707) entweder dem E-BUS-MASTER oder der Zustandsmaschine (0703) zuordnet. Das MODE-PLUREG (0708) wird über den Ladelogikbus (0709) oder den I-BUS (0713) eingestellt und konfiguriert den Adresszähler (0710), sowie die Zustandsmaschine (z.B. ausmaskieren der INPUT-CELL). Werden Daten des E-BUS (0705) in das E-GATE-REG (0707) gespeichert, wird der Zugriff im SET-REG (0712) vermerkt. Der Zustand von 0712 ist über ein Tor (0715), dessen Steuerung die selbe wie beim Latch (0707) ist, am E-BUS abfragbar. Ein lesender Zugriff, das E-GATE 0711 ist aktiviert und die Daten gelangen auf den I-BUS (0713) setzt 0712 über 0717 zurück. Alternativ kann 0712 über die Zustandsmaschine (0703) zurückgesetzt werden (0718). Die von 0710 generierten Adressen werden über das Tor (ADR-GATE) 0716 auf den E-BUS übertragen, 0716 wird von der Zustandsmaschine (0703) aktiviert, wenn diese E-BUS-MASTER ist. Es besteht die Möglichkeit die INPUT-CELL statt über die Ladelogik über den Baustein (DFP, FPGA, DPGA, o.ä.) selbst dynamisch umzukonfigurieren und zu steuern. Hierzu dient der der I-BUS-Anschluß an das EB-REG (0702) und das MODE-PLUREG (0708).

**Figur 8** zeigt das MODE-PLUREG (0801) einer INPUT- oder OUTPUT-CELL, das über den Ladelogikbus (0802) von der Ladelogik oder über einen I-BUS (0808) beschrieben wird. Das jeweilige Bussystem wird über den Multiplexer (0809) ausgewählt, dabei ist die Steuerung des Multiplexers nicht eingezeichnet, da eine gewöhnliche Dekodierlogik verwendet werden kann. Die Zählereinstellungen, wie Schrittbreite, Zählrichtung und Freischaltung des Zählers werden direkt (0807) an den

Zähler (0803) geleitet. Die Grundadresse kann entweder über einen Load (0804) direkt in den Zähler geschrieben (0805) werden, oder in einem Zusatz (0811) zu 0801 zwischengespeichert werden. Die für die Zustandsmaschine relevanten Einträge in 0801 gelangen über ein Tor (0806), das von der Zustandsmaschine für die jeweilig aktivierte INPUT- oder OUTPUT-CELL geöffnet wird, zur Zustandsmaschine.

**Figur 9a** zeigt eine Businterfaceschaltung mit Zustandsmaschine (0901), MASTER-GATE (0902) und EB-REG (0903). Die INPUT-CELLs (0904) transferieren Daten vom E-BUS (0905) auf den II-BUS (0906). Die OUTPUT-CELLs (0907) transferieren Daten vom IO-BUS (0908) auf den E-BUS (0905). Über den Steuerbus (0909) sind sämtliche Baugruppen miteinander verbunden.

**Figur 9b** zeigt eine Businterfaceschaltung mit Zustandsmaschine (0901), MASTER-GATE (0902) und EB-REG (0903). Die INPUT-CELLs (0904) transferieren Daten vom E-BUS (0905) auf den bidirektionalen I-BUS (0910). Die OUTPUT-CELLs (0907) transferieren Daten vom bidirektionalen I-BUS (0910) auf den E-BUS (0905). Über den Steuerbus (0909) sind sämtliche Baugruppen miteinander verbunden. Es sind auch Interfaceschaltungen denkbar, die beide Möglichkeiten (Fig. 9a und 9b) gemischt verwenden.

**Figur 10a** zeigt zwei Bausteine (DFPs, FPGAs, DPGAs, o.ä.) (1001) über den E-BUS (1002) miteinander vernetzt.

**Figur 10b** zeigt die Vernetzung einer Mehrzahl von Bausteinen (DFPs, FPGAs, DPGAs, o.ä.) (1001) über den E-BUS (1002).

**Figur 10c** zeigt die Vernetzung einer Mehrzahl von Bausteinen (DFPs, FPGAs, DPGAs, o.ä.) (1001) über den E-BUS (1002). Die Vernetzung kann zu einer Matrix ausgebaut werden. Ein Baustein (1001) kann auch eine Mehrzahl von Bussystemen (1002) verwalten.

**Figur 10d** zeigt die Vernetzung eines Bausteins (DFPs, FPGAs, DPGAs, o.ä.) (1001) mit einem Speicherbaustein oder einer Speicherbank (1003) über den E-BUS (1002).

**Figur 10e** zeigt die Vernetzung eines Bausteins (DFPs, FPGAs, DPGAs, o.ä.) (1001) mit einem Peripheriebaustein oder einer Peripheriegruppe (1004) über den E-BUS (1002).

**Figur 10f** zeigt die Vernetzung eines Bausteins (DFPs, FPGAs, DPGAs, o.ä.) (1001) mit einem Speicherbaustein oder einer Speicherbank (1003) und mit einem Peripheriebaustein oder einer Peripheriegruppe (1004) über den E-BUS (1002).

**Figur 10g** zeigt die Vernetzung eines Bausteins (DFPs, FPGAs, DPGAs, o.ä.) (1001) mit einem Speicherbaustein oder einer Speicherbank (1003) und mit einem Peripheriebaustein oder einer Peripheriegruppe (1004) und einem weiteren Baustein (DFPs, FPGAs, DPGAs, o.ä.) (1001) über den E-BUS (1002).

**Figur 11** zeigt den Aufbau des EB-REG. Die Bussysteme E-BUS (1103); Ladelogikbus (1104), über welchen die Ladelogik Zugriff auf das EB-REG hat und der lokale interne Bus zwischen den INPUT-/OUTPUT-CELLs, der Zustandsmaschine und dem EB-REG (1105, vgl. 0407, 0417, 0517), sowie möglicherweise ein I-BUS (1114) werden auf einen Multiplexer (1106) geführt. Der Multiplexer (1106) wählt entweder einen der Busse oder die Rückkopplung auf das Register (1108) aus und schaltet die Daten zum Eingang des Registers (1108) durch. Das MASTER-Bit wird getrennt über den Multiplexer (1107) zum Register (1108) geführt. Der Multiplexer wird über das RESET-Signal (1101) (Rücksetzen oder Initialisieren des Bausteines) gesteuert. Liegt ein RESET an, so schaltet der Multiplexer (1107) das Signal eines externen Chipanschlusses (1102) auf den Eingang des Registers (1108) durch, andernfalls wird der Ausgang des Multiplexers (1106) auf den Eingang des Registers (1108) durchgeschaltet. Dadurch kann MASTER vorbelegt werden. Das Register (1108) wird über den Systemtakt (1112) getaktet. Der Inhalt des Registers (1108) wird über ein Tor (1109, 1110, 1111, 1113) auf das jeweils lesend zugreifenden Bussystem (1103, 1104, 1105, 1114) geschaltet. Die Steuerung der Tore (1109, 1110, 1111, 1113), sowie des Multiplexers (1106) ist nicht eingezeichnet, da eine gewöhnliche Dekodierlogik verwendet werden kann.

## 5 Ausführungsbeispiele

**Figur 12** zeigt ein Beispiel bei dem das Standardbussystems RAMBUS (1203) eingesetzt wird. Ein Baustein (DFPs, FPGAs, DPGAs, o.ä.) (1201) ist über das Bussystem (1203) mit weiteren Baugruppen (Speicher, Peripherie, weitere DFPs, FPGAs, DPGAs, o.ä.) (1202) verbunden. Der Baustein (1201) kann unabhängig vom Bussystem (1203) weitere Anschlußleitungen (1204), z.B. wie nach dem Stand der Technik üblich, zum Anschluß beliebiger Schaltkreise aufweisen.

## Patentansprüche

- 1 : Bussystem, hergestellt durch Bündelung mehrere einzelner Leitungen oder Busse oder Teilbusse (vgl. Fig. 4/5) innerhalb eines Bausteines der Gattung DFP, FPGA, DPGA, sowie allen Bausteinen mit zwei- oder mehrdimensionaler programmierbare Zellstruktur (vgl. Fig. 1/2/3), über welches die Bausteine zu Mehreren zusammengefaßt werden können und/oder Speicher und/oder Peripherie anschließbar ist (vgl. Fig. 10).
- 2 : Bussystem nach Anspruch 1, dadurch gekennzeichnet, daß eine oder mehrere Schnittstellen (Fig. 6/7) die Zusammenfassung der Leitungen übernehmen und das Bussystem erzeugen.
- 3 : Bussystem nach Anspruch 1 und 2, dadurch gekennzeichnet, daß eine oder mehrere Zustandsmaschine(n) (0703/0603) die Schnittstellen (vgl. Fig. 6/7) steuert/steuern.
- 4 : Bussystem nach Anspruch 1, 2 und 3, dadurch gekennzeichnet, daß die Zustandsmaschine auch den externen Bus steuert.
- 5 : Bussystem nach Anspruch 1 und 2, dadurch gekennzeichnet, daß ein Adressgenerator existiert (0610/0710), der die Adressen für die über den Bus zu kontaktierenden Bausteine generiert.
- 6 : Bussystem nach Anspruch 1 und 2, dadurch gekennzeichnet, daß die Schnittstellen ein internes Bussystem oder mehrere interne Bussysteme, das/die aus mehreren Leitungen zusammengefaßt sein kann/können (vgl. Fig. 4/5), zum Schreiben und Lesen verwendet (vgl. Fig. 9a, I-BUS).
- 7 : Bussystem nach Anspruch 1 und 2, dadurch gekennzeichnet, daß die Schnittstellen ein internes Bussystem oder mehrere interne Bussysteme, das/die aus mehreren Leitungen zusammengefaßt sein kann/können (vgl. Fig. 4/5), entweder zum Schreiben oder Lesen verwendet (vgl. Fig. 9b, II-BUS, IO-BUS).
- 8 : Bussystem nach Anspruch 1 und 2, dadurch gekennzeichnet, daß die Schnittstellen ein internes Bussystem oder mehrere interne Bussysteme, das/die aus mehreren Leitungen zusammengefaßt sein kann/können (vgl. Fig. 4/5), nach Anspruch 6 und 7 gemischt betreiben.
- 9 : Bussystem nach Anspruch 1 und 2, dadurch gekennzeichnet, daß ein Register zur Verwaltung und Steuerung des Bussystems existiert (EB-REG, 0702, 0602).
- 10 : Bussystem nach Anspruch 1 und 2, dadurch gekennzeichnet, daß die Bussteuerung durch einen Baustein (E-BUS-MASTER) erfolgt, der auf eine Vielzahl von untergeordneten Bausteinen (E-BUS-SLAVE) zugreift.
- 11 : Bussystem nach Anspruch 1, 2 und 10, dadurch gekennzeichnet, daß die Bussteuerung dynamisch von einem Baustein (E-BUS-MASTER) an einen anderen übergeben wird (MASTER-Eintrag im EB-REG).
- 12 : Bussystem nach Anspruch 1, 2, 10 und 11, dadurch gekennzeichnet, daß ein untergeordneter Baustein (E-BUS-SLAVE) den Erhalt der Buskontrolle fordern kann (REQ-MASTER-Eintrag im EB-REG).
- 13 : Bussystem nach Anspruch 1 und 2, dadurch gekennzeichnet, daß ein Register existiert, das anzeigt, ob Daten in den Schnittstellen gespeichert sind (SET-REG, 0612, 0712).
- 14 : Bussystem nach Anspruch 1 und 2, dadurch gekennzeichnet, daß die Schnittstellen entweder direkt auf dem Baustein implementiert sind, oder durch die Konfiguration von Logikzellen, das sind Zellen in DFP-, FPGA-, DPGA, oder ähnlichen Bausteinen, die einfache logische oder arithmetische

Aufgaben gemäß ihrer Konfiguration erfüllen, entstehen.

- 15 : Bussystem nach Anspruch 1 und 2, dadurch gekennzeichnet, daß die Schnittstellen von einer Ladelogik und/oder dem Baustein selbst (vgl. Fig. 8/11) konfiguriert werden können.
- 16 : Bussystem nach Anspruch 1, 2 und 15, dadurch gekennzeichnet, daß die Ladelogik partiell auf dem Baustein integriert ist.
- 17 : Bussystem nach Anspruch 1 und 2, dadurch gekennzeichnet, daß Standardbussysteme eingesetzt werden können (vgl. Fig. 12).
- 18 : Bussystem nach Anspruch 1 und 2, dadurch gekennzeichnet, daß der Baustein weitere gewöhnliche Anschlüsse nach bei DFPs, FPGAs, DP-GAs, o.ä. üblicher Art besitzt (vgl. Fig. 12 1201, 1204).



## 6 Begriffsdefinition

<b>ADR-GATE</b>	Tor, das die Adressen auf den E-BUS aufschaltet, sofern der Baustein E-BUS-MASTER ist.
<b>DFP</b>	Datenflußprozessor nach Patent/Offenlegung DE 44 16 881
<b>DPGA</b>	Dynamisch konfigurierbare FPGAs. Stand der Technik
<b>D-FlipFlop</b>	Speicherelement, welches ein Signal bei der steigenden Flanke eines Taktes speichert.
<b>EB-REG</b>	Register, das die Statussignale zwischen I-BUS und E-BUS speichert.
<b>E-BUS</b>	Externes Bussystem außerhalb eines Bausteines.
<b>E-BUS-MASTER</b>	Baustein, der den E-BUS steuert. Aktiv.
<b>E-BUS-SLAVE</b>	Baustein, der vom E-BUS-MASTER angesprochen wird. Passiv.
<b>E-GATE</b>	Tor, das von der bausteininternen Zustandsmaschine oder vom E-BUS-MASTER gesteuert wird und Daten auf den E-BUS aufschaltet.
<b>E-GATE-REG</b>	Register in das die Daten eingetragen werden, die über das E-GATE auf den E-BUS übertragen werden.
<b>E-READ</b>	Flag im EB-REG, das anzeigt, daß die OUTPUT-CELLs komplett auf den E-BUS übertragen wurden.
<b>E-WRITE</b>	Flag im EB-REG, das anzeigt, daß der E-BUS komplett in die INPUT-CELLs übertragen wurde.
<b>Flag</b>	(Fahne). Statusbit in einem Register, das einen Zustand anzeigt.
<b>FPGA</b>	Programmierbarer Logikbaustein. Stand der Technik.
<b>Handshake</b>	Signalprotokoll, bei dem ein Signal A einen Zustand anzeigt und ein anderes Signal B bestätigt, daß es Signal A akzeptiert und darauf reagiert (hat).
<b>INPUT-CELL</b>	Baugruppe, die Daten vom E-BUS auf einen I-BUS überträgt.
<b>I-BUS<sub>n</sub></b>	(auch <i>I-BUS</i> ) Bausteininternes Bussystem, das auch aus Bündeln von einzelnen Leitungen bestehen kann. n gibt die Nummer des Busses an.
<b>II-BUS<sub>n</sub></b>	(auch <i>II-BUS</i> ) Bausteininternes Bussystem, das auch aus Bündeln von einzelnen Leitungen bestehen kann. Der Bus wird von einer INPUT-CELL getrieben und geht auf logische Eingänge. n gibt die Nummer des Busses an.
<b>IO-BUS<sub>n</sub></b>	(auch <i>IO-BUS</i> ) Bausteininternes Bussystem, das auch aus Bündeln von einzelnen Leitungen bestehen kann. Der Bus wird von logischen Ausgängen getrieben und geht an eine OUTPUT-CELL. n gibt die Nummer des Busses an.

<b>I-GATE</b>	Tor, das Daten auf den I-BUS aufschaltet.
<b>I-GATE-REG</b>	Register, das von der bausteininternen Zustandsmaschine oder vom E-BUS-MASTER gesteuert wird und in das die Daten eingetragen werden, die über das I-GATE auf den I-BUS übertragen werden.
<b>I-READ</b>	Flag im EB-REG, das anzeigt, daß die INPUT-CELLs komplett auf den I-BUS übertragen wurden.
<b>I-WRITE</b>	Flag im EB-REG, das anzeigt, daß der I-BUS komplett in die OUTPUT-CELLs übertragen wurde.
<b>Kantenzelle</b>	Zelle am Rand eines Zellarrays, oftmals mit direktem Kontakt zu den Anschlüssen eines Bausteines.
<b>Konfigurieren</b>	Einstellen der Funktion und Vernetzung einer logischen Einheit, einer (FPGA)-Zelle (Logikzelle) oder einer PAE (vgl. umkonfigurieren).
<b>Ladelogik</b>	Einheit zum Konfigurieren und Umkonfigurieren einer PAE oder Logikzelle. Ausgestaltet durch einen speziell an seine Aufgabe angepaßten Mikrokontroller.
<b>Latch</b>	Speicherelement, das ein Signal für gewöhnlich während des H-Pegels transparent weiterleitet und während des L-Pegels speichert. In PAEs werden teilweise Latches gebraucht, bei denen die Funktion der Pegel genau umgekehrt ist. Hierbei wird vor den Takt eines üblichen Latch ein Inverter geschaltet.
<b>Logikzellen</b>	Bei DFPs, FPGAs, DPGAs verwendete konfigurierbare Zellen, die einfache logische oder arithmetische Aufgaben gemäß ihrer Konfiguration erfüllen.
<b>MASTER</b>	Flag im EB-REG, das anzeigt, daß der Baustein E-BUS-MASTER ist.
<b>MODE-PLUREG</b>	Register in dem die Ladelogik die Konfiguration einer INPUT-/OUTPUT-CELL einstellt.
<b>OUTPUT-CELL</b>	Baugruppe, die Daten von einem I-BUS auf den E-BUS überträgt.
<b>PAE</b>	Processing Array Element: EALU mit O-REG, R-REG, R2O-MUX, F-PLUREG, M-PLUREG, BM-, SM-, Sync-, StateBack- und Power-UNIT.
<b>PLU</b>	Einheit zum Konfigurieren und Umkonfigurieren einer PAE oder Logikzelle. Ausgestaltet durch einen speziell an seine Aufgabe angepaßten Mikrokontroller.
<b>REQ-MASTER</b>	Flag im EB-REG, das anzeigt, daß der Baustein E-BUS-MASTER werden möchte.
<b>RS-FlipFlop</b>	Reset-/Set-FlipFlop. Speicherelement, das durch 2 Signale umgeschaltet werden kann.

- SET-REG** Register das anzeigt, daß Daten in ein I-GATE-REG oder E-GATE-REG geschrieben und noch nicht gelesen wurden.
- STATE-GATE** Tor, das den Ausgang des SET-REG auf den E-BUS schaltet.
- Tor** Schalter, der ein Signal weiterleitet oder sperrt. Einfacher Vergleich: Relais
- Umkonfigurieren** Neues Konfigurieren von einer beliebigen Menge von PAEs oder Logikzellen während eine beliebige Restmenge von PAEs oder Logikzellen ihre eigenen Funktionen fortsetzen (vgl. konfigurieren).
- Zustandsmaschine** Logik, die diversen Zuständen annehmen kann. Die Übergänge zwischen den Zuständen sind von verschiedenen Eingangsparametern abhängig. Diese Maschinen werden zur Steuerung komplexer Funktionen eingesetzt und entsprechen dem Stand der Technik

## 7 Konventionen

### 7.1 Namenskonvention

Baugruppe	-UNIT
Betriebsart	-MODE
Multiplexer	-MUX
Negiertes Signal	not-
Register für PLU sichtbar	-PLUREG
Register intern	-REG
Schieberegisters	-sft

### 7.2 Funktionskonvention

Schieberegisters sft

UND-Funktion &

A	B	Q
0	0	0
0	1	0
1	0	0
1	1	1

ODER-Funktion #

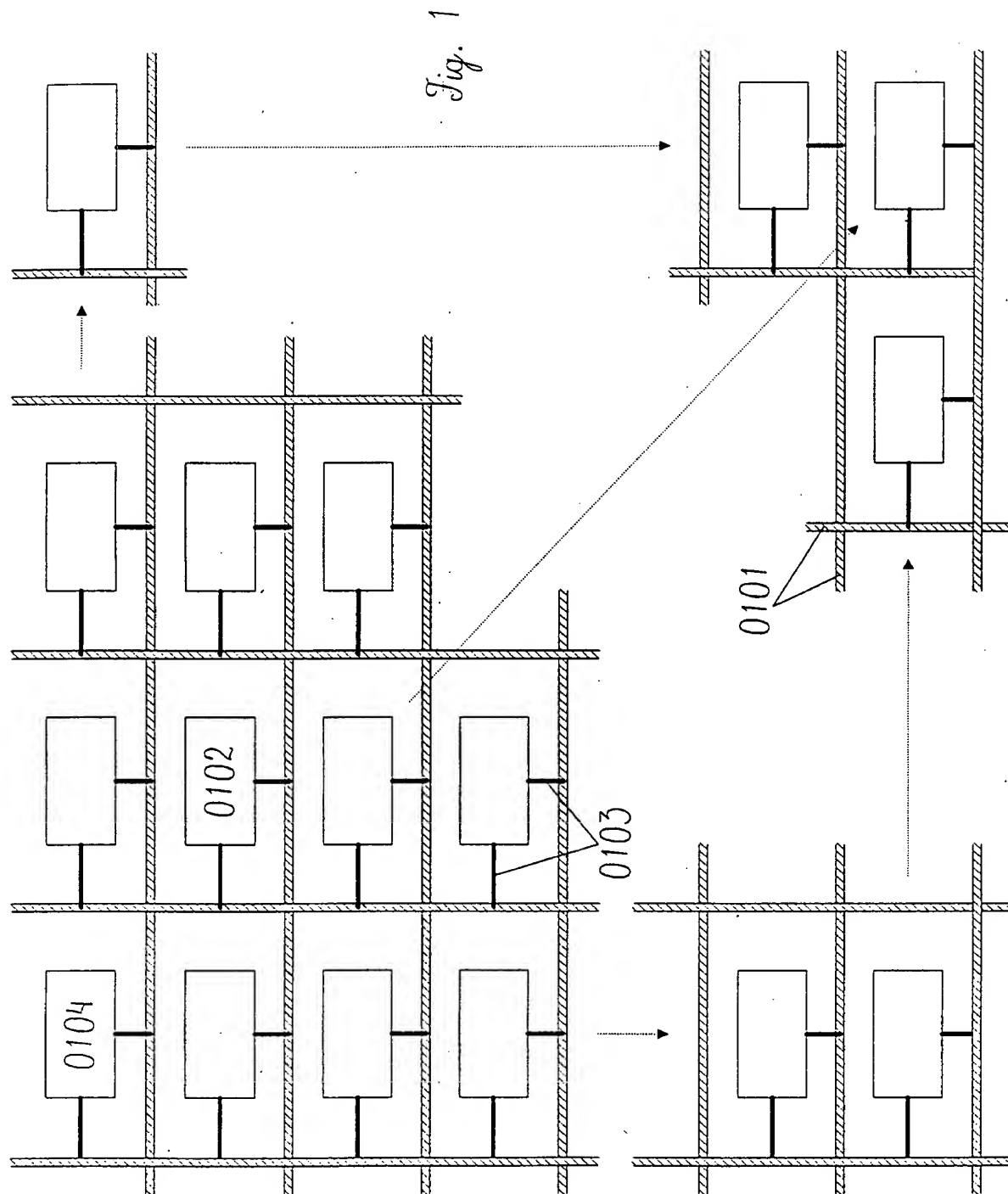
A	B	Q
0	0	0
0	1	1
1	0	1
1	1	1

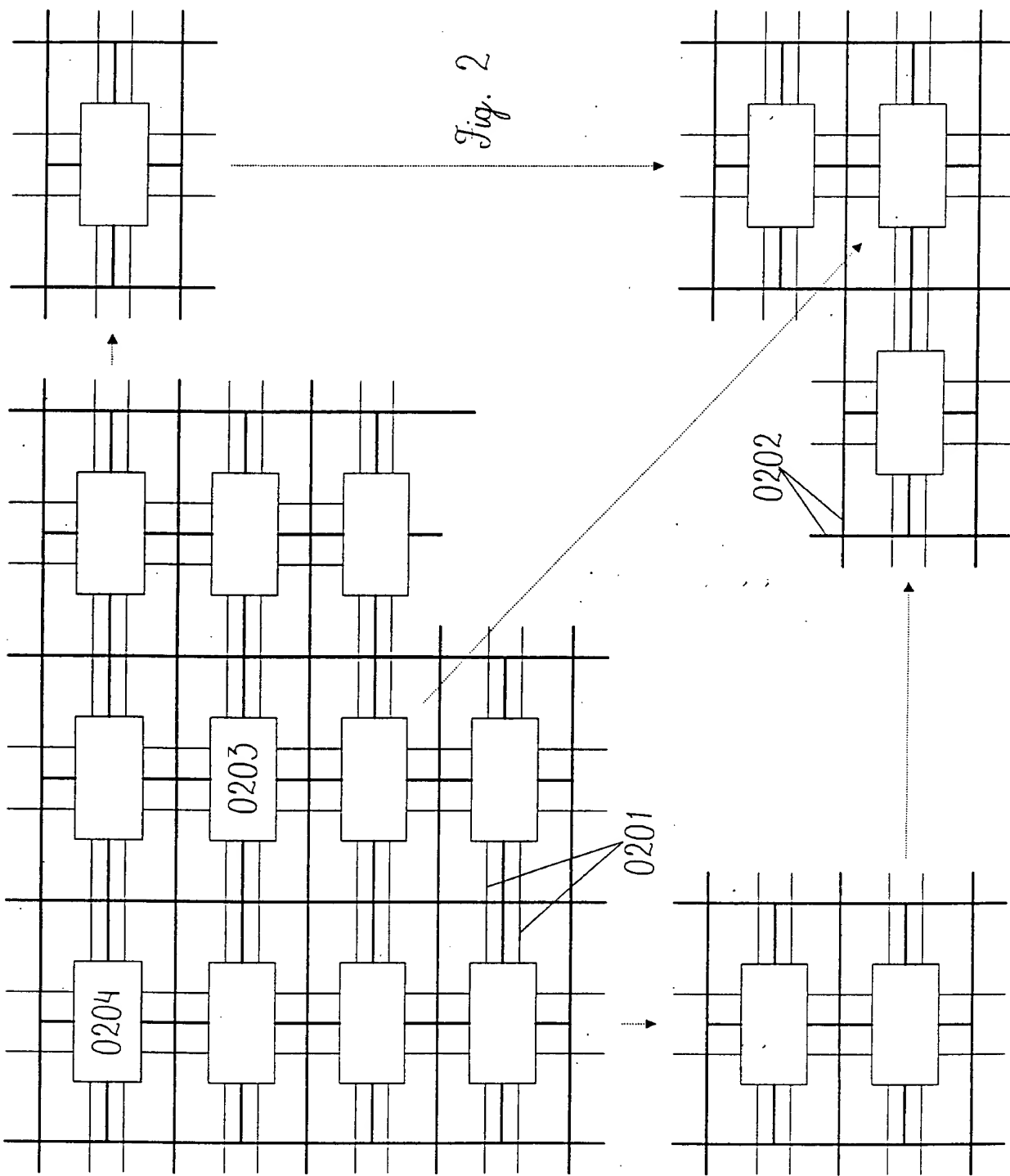
NICHT-Funktion !

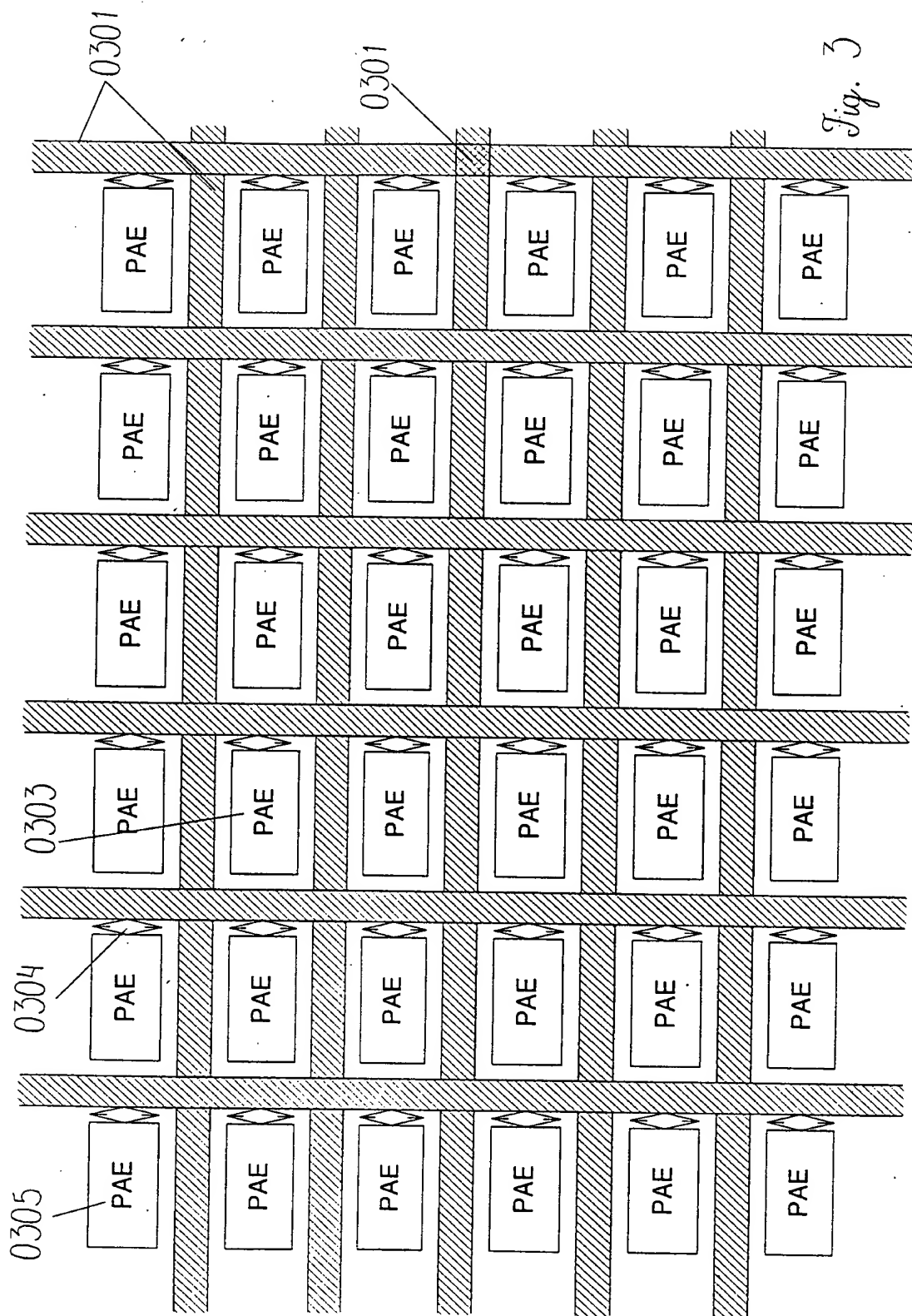
A	Q
0	1
1	0

**TOR-Funktion**      G

EN	D	Q
0	0	-
0	1	-
1	0	0
1	1	1









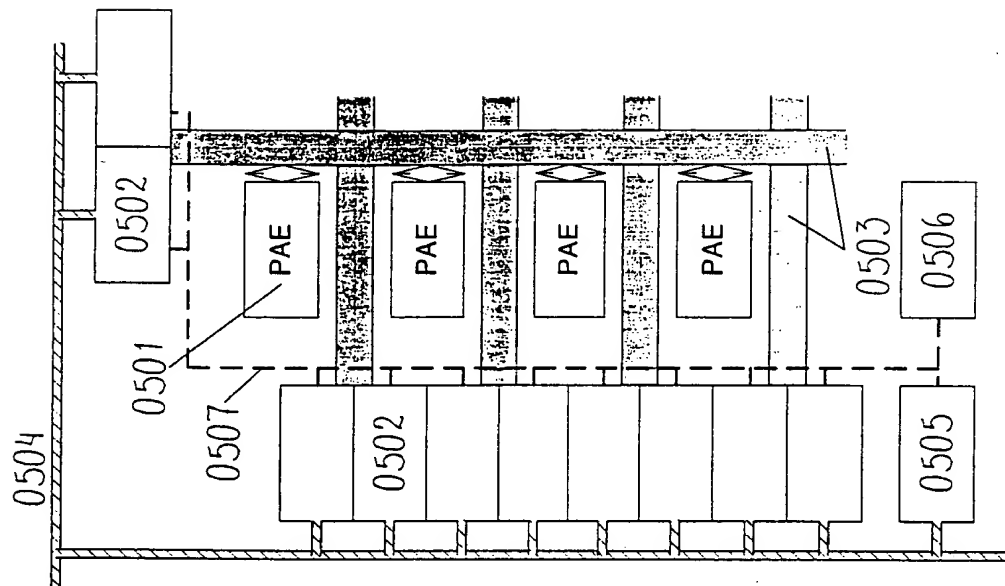


Fig. 5

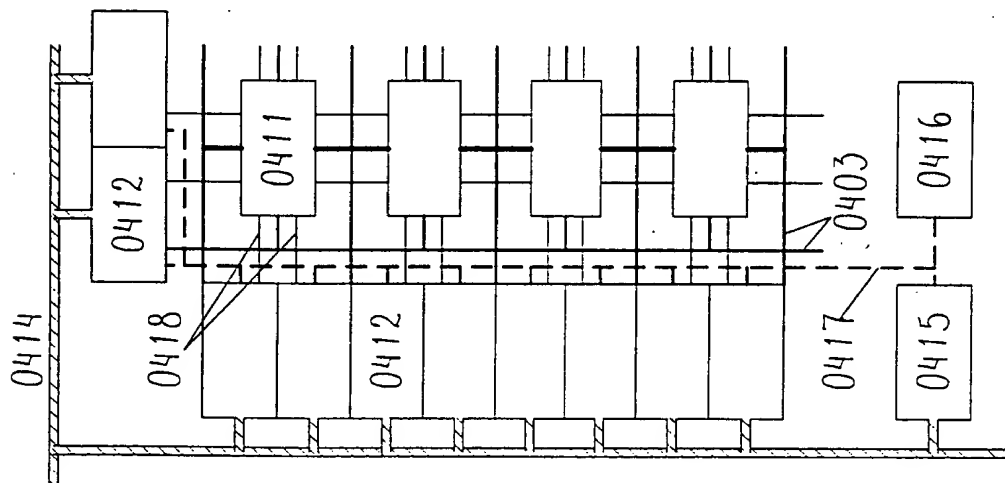


Fig. 4b

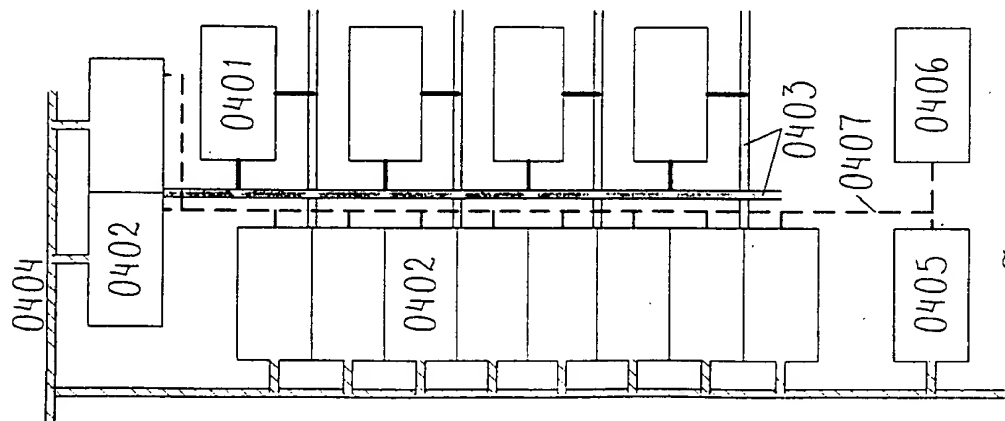


Fig. 4a

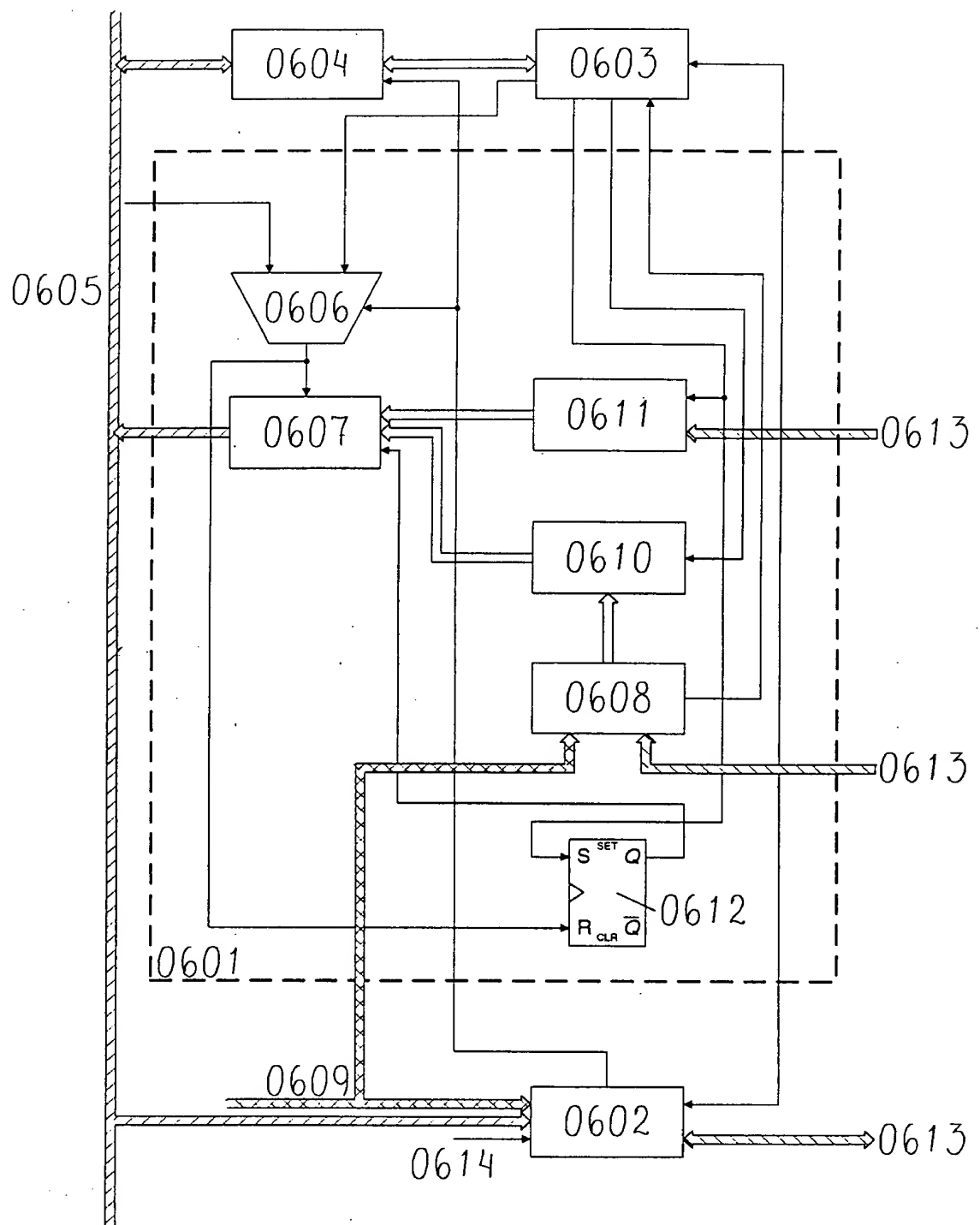
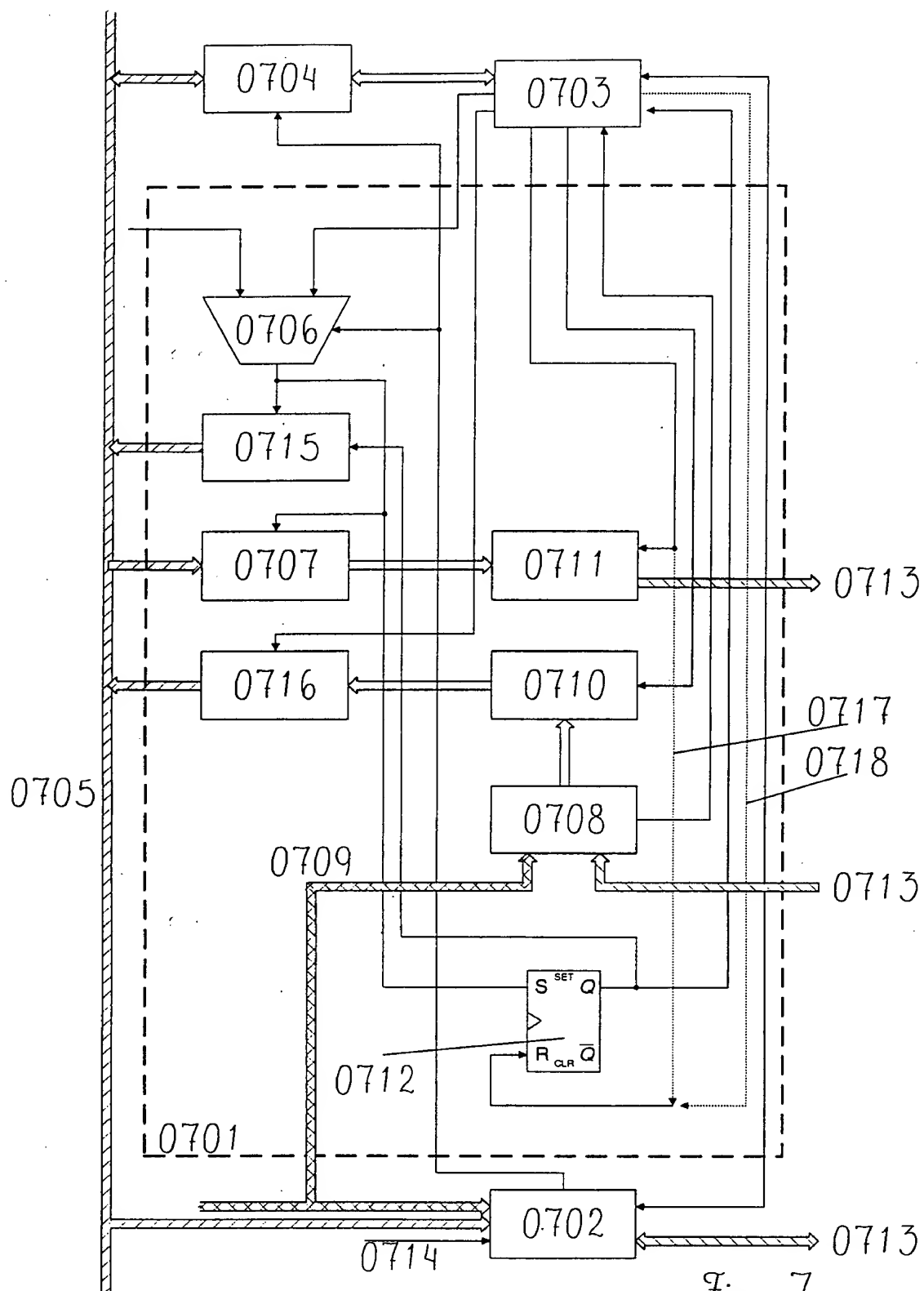


Fig. 6



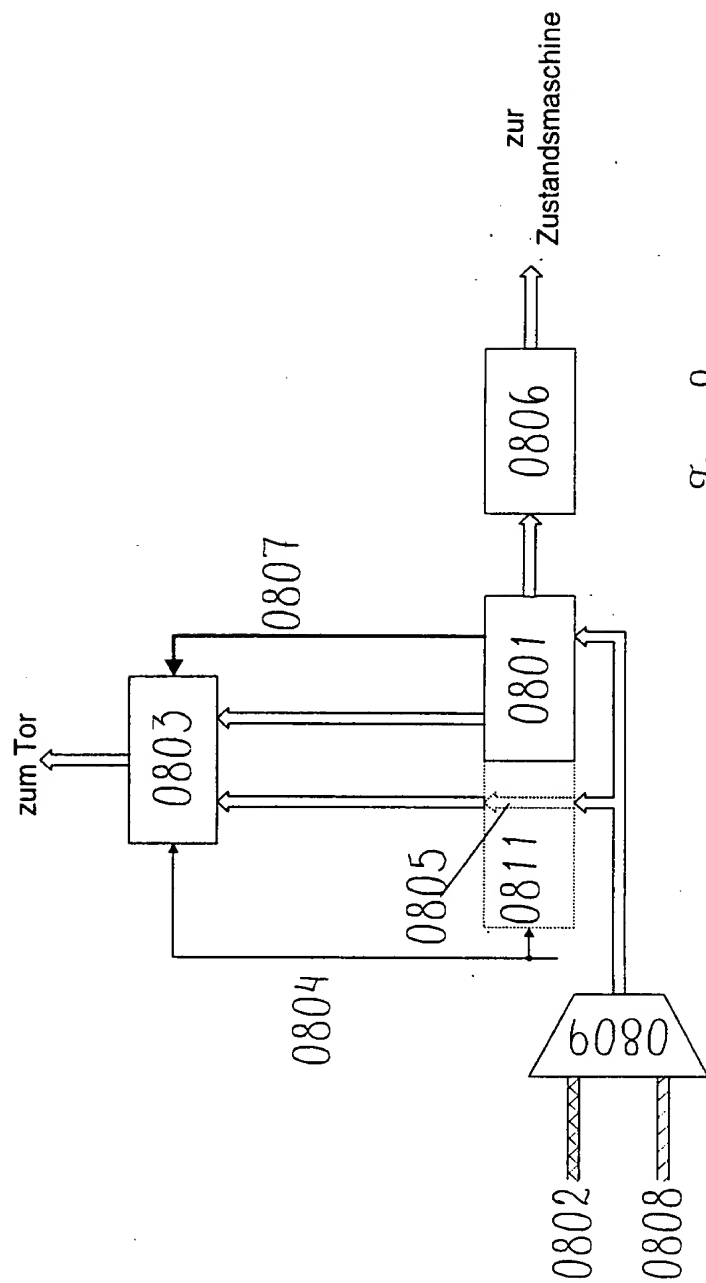


Fig. 8

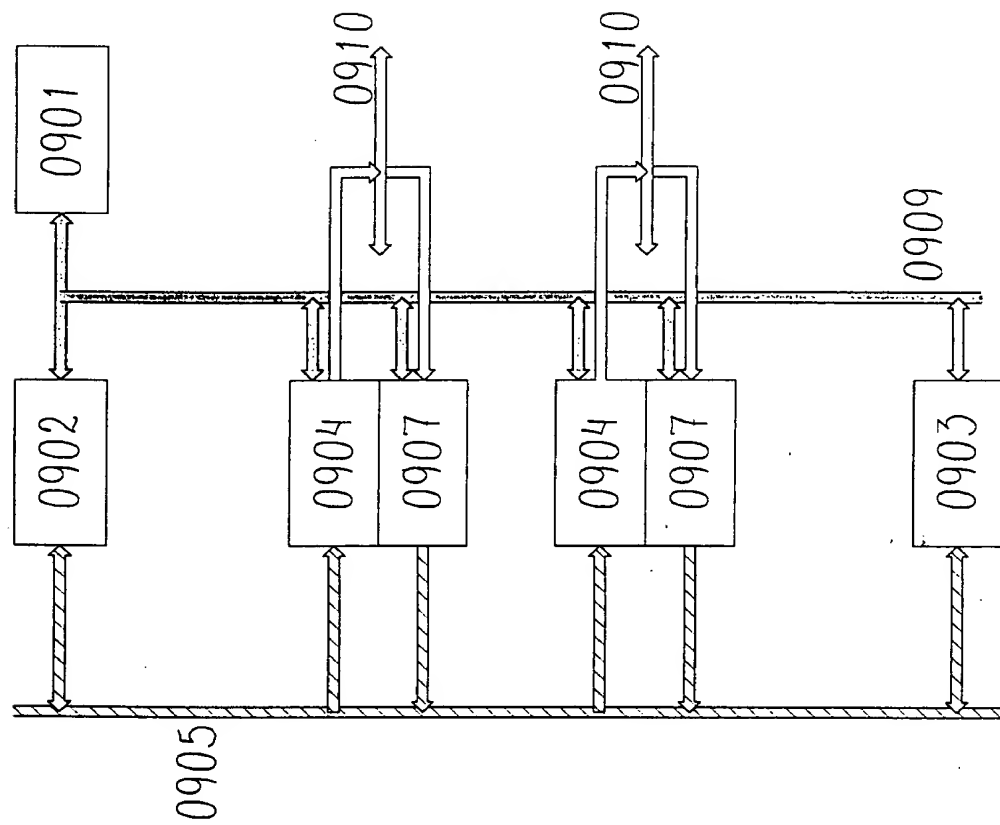


Fig. 9a

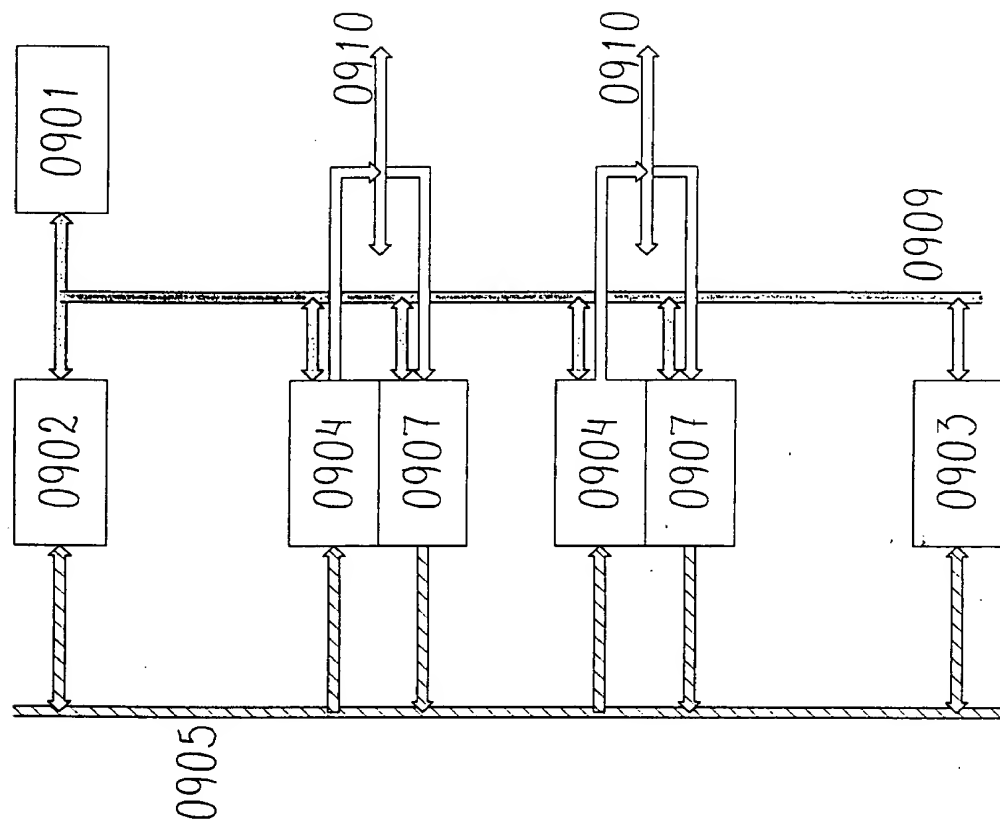
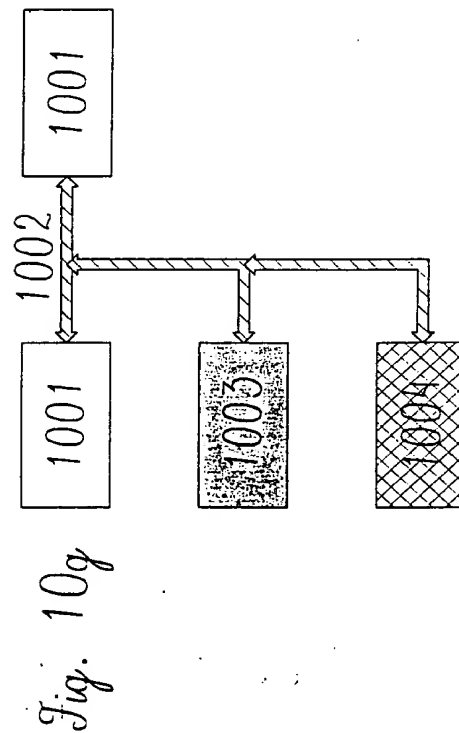
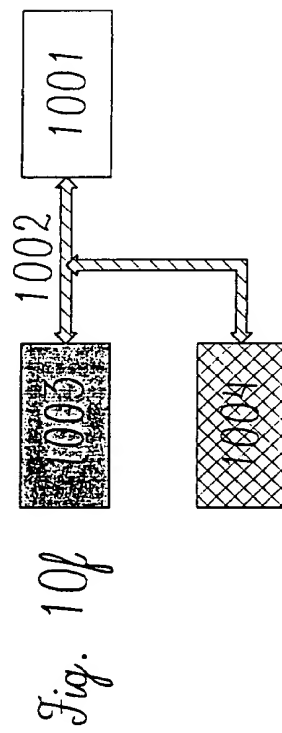
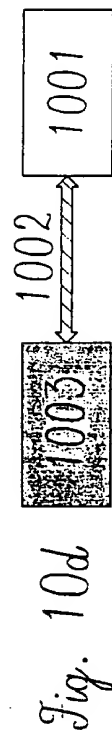
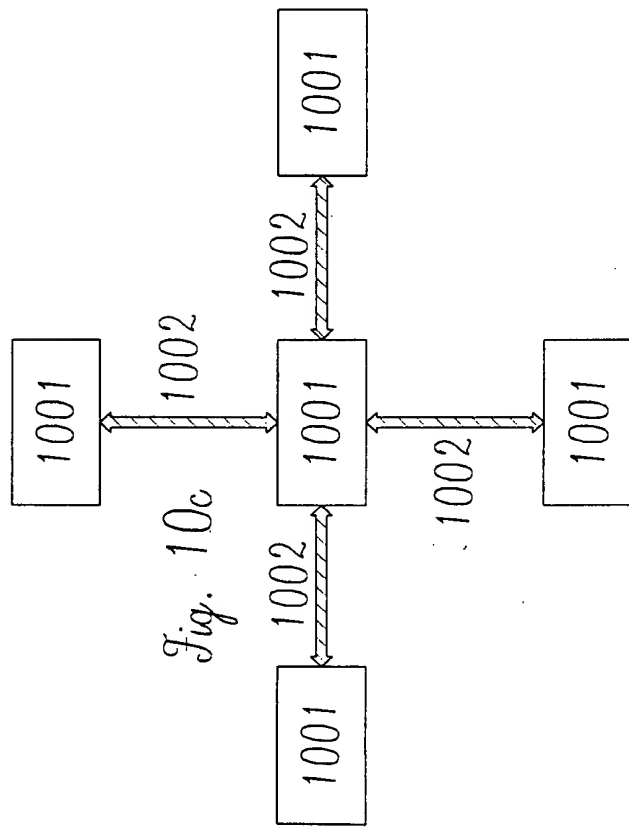
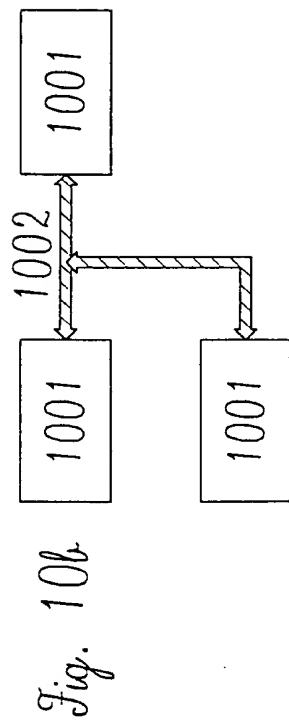
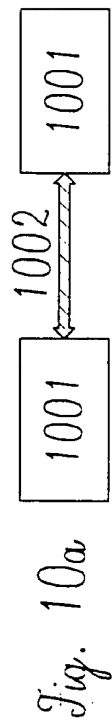


Fig. 9b



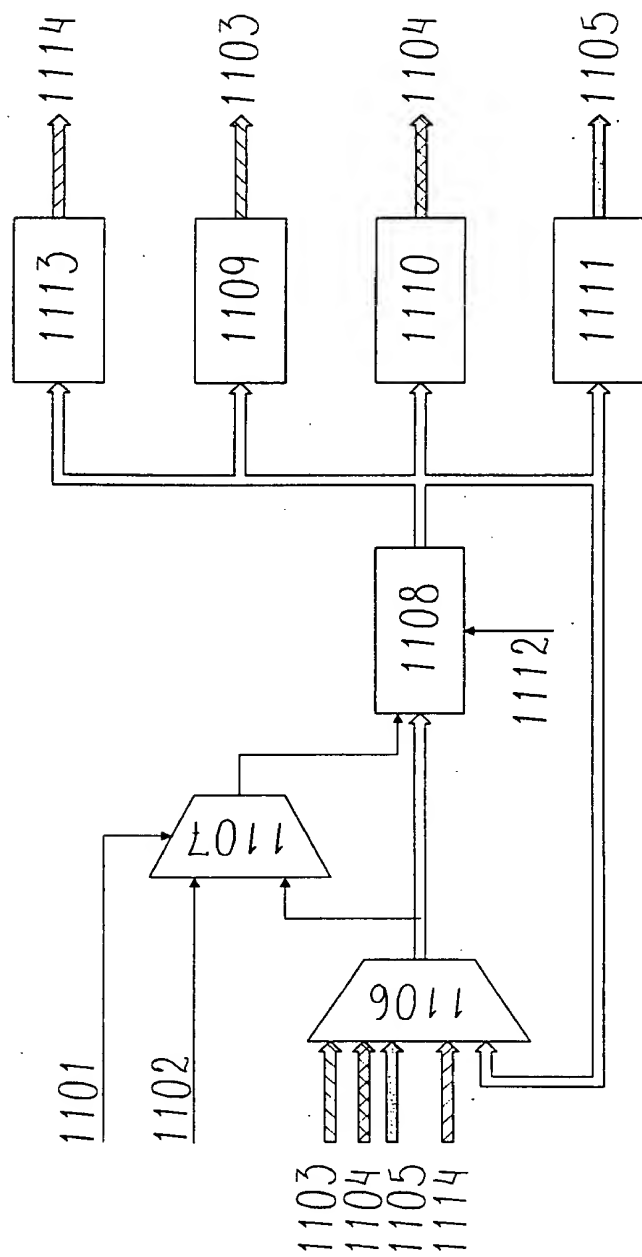


Fig. 11

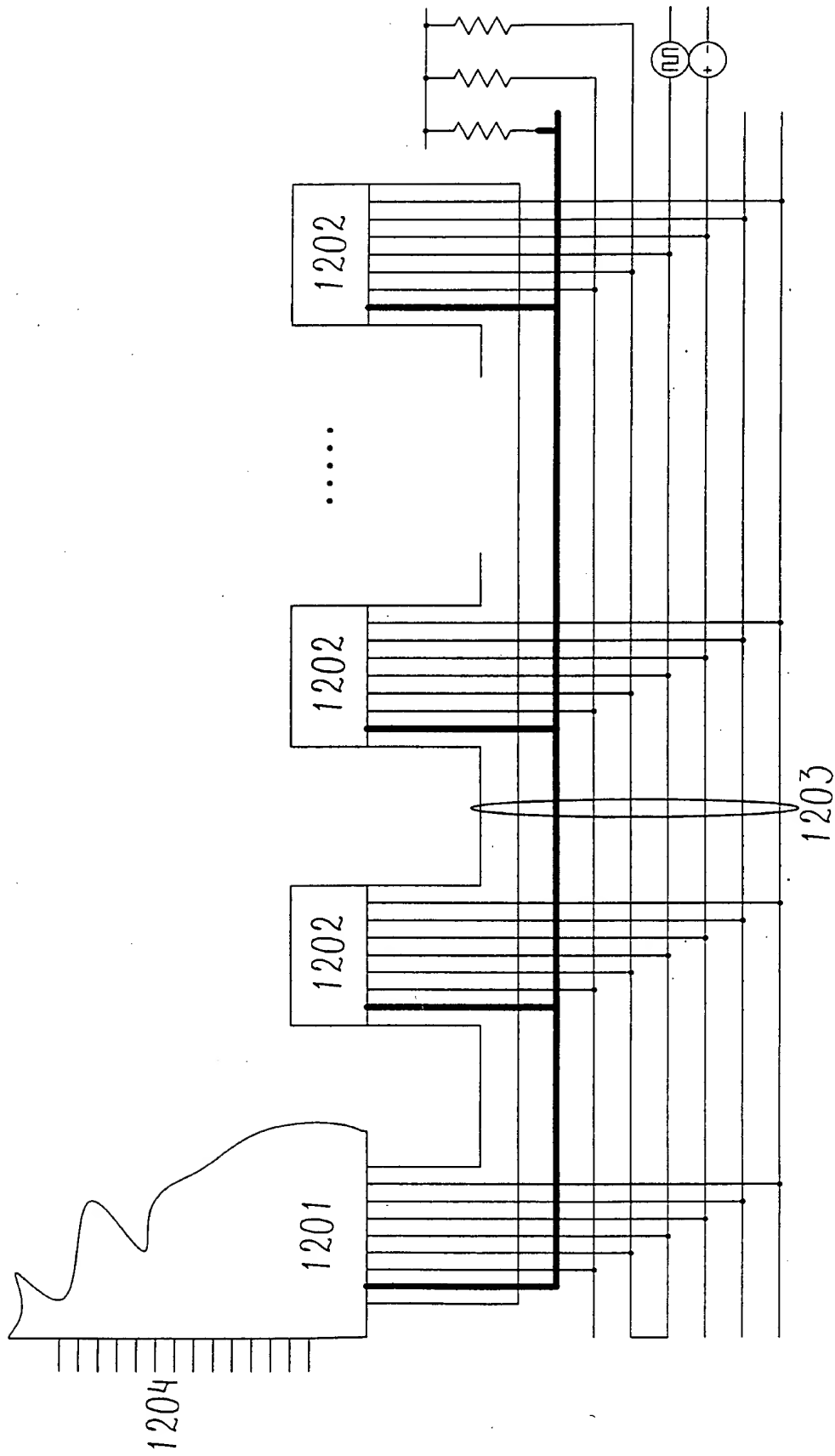


Fig. 12